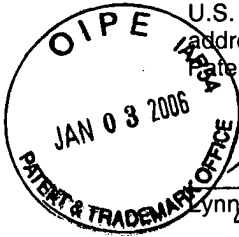


CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the U.S. Postal Service for First-Class Mail Delivery, postage prepaid, addressed to: Mail Stop Appeal Brief – Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on 29 December 2005.



Lynne M. Milliot
Lynne M. Milliot

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Muljadi SULISTIO et al.

Application No. 10/026,681

Confirmation No. 7035

Filed: 18 December 2001

Title: **METHOD AND APPARATUS FOR
GENERIC SEARCH INTERFACE ACROSS
DOCUMENT TYPES**

Group Art Unit: 2176

Examiner: Robert STEVENS

CUSTOMER NO. 22470

MAIL STOP APPEAL BRIEF - PATENTS

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

APPEAL BRIEF

Sir:

This Appeal Brief is filed in support of Appellants' appeal from the Final Office Action, mailed 29 June 2005, which was followed by an Advisory Action, mailed 26 September 2005. A Notice of Appeal was filed on 29 September 2005. An appropriate request for an extension of time accompanies this document.

The appropriate fee as set forth in § 41.20 (b)(2) of \$500.00 is covered in the enclosed check. Should it be determined that additional fees are required, the Commissioner is hereby authorized to charge those fees to Deposit Account No. 50-0869 (Attorney Docket No. JGR 1009-1).

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST	1
II. RELATED APPEALS AND INTERFERENCES	1
III. STATUS OF CLAIMS	1
IV. STATUS OF AMENDMENTS.....	1
V. SUMMARY OF CLAIMED SUBJECT MATTER	1
VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL	3
VII. ARGUMENT	3
A. Preliminary Review of the Technology Disclosed and References.....	4
1. The Disclosed Technology.....	4
2. The Probst Reference.....	5
3. The Harold Reference	6
4. The XPath Spec Reference	6
B. Rejection of Independent Claim 1 Under Section 103(a) as Unpatentable over Probst in view of Harold was Improper	7
C. Rejection of Independent Claim 8 Under Section 103(a) as Unpatentable over Probst in view of Harold was Improper	8
D. Rejection of Independent Claim 24 Under Section 103(a) as Unpatentable over Probst in view of Harold was Improper	10
CONCLUSION.....	11
CLAIMS APPENDIX	12

I. REAL PARTY IN INTEREST

The real party in interest is Open Invention Network, the assignee of record. The status of recording of the most recent assignment is unclear.

II. RELATED APPEALS AND INTERFERENCES

There are no known appeals or interferences relating to this case.

III. STATUS OF CLAIMS

Claims 1-24 are pending in this application. All have been rejected and all of the rejections are subject to this appeal. All amendments have been entered.

Claims 1, 3-8, 10-16 and 18-24 were rejected under 35 U.S.C. 103(a) as being unpatentable over Probst et al., U.S. Patent Application Publication No. 2003/0140034 ("Probst"), in view of Elliotte Rusty Harold, "XML: Extensible Markup Language," IDG Books Worldwide, Inc., Foster City, CA (1998) ("Harold").

Claims 2, 9 and 17 were rejected under 35 U.S.C. 103(a) as being unpatentable over Probst, in view of Harold, and in further view of XML Path Language (XPath) Version 1.0 (W3C Recommendation, 16 November 1999, hereafter "XPath Spec").

IV. STATUS OF AMENDMENTS

A substitute specification was entered, which moved XML code excerpts to a CD-ROM appendix. However, the original specification ("OSpec") is easier to use for present purposes and is referenced in this brief. (If the amended specification is used, note that the paragraph numbering after [0004] is off by +1.)

All amendments to the claims have been entered.

V. SUMMARY OF CLAIMED SUBJECT MATTER

Independent claims 1, 8 and 24, are addressed individually. The remaining claims stand or fall with the independent claims from which they depend:

Claim 1 presents a computer-implemented method of searching a plurality of self-describing, structured documents, said documents including document fields. (OSpec, [0039], [0065]-0070)) The method includes three phases. First, providing a graphical user interface (e.g., FIG. 5) including a document type selection filter (518); one or more document field selection filters (512), context sensitive to a selected

document type; one or more value specification fields (514), context sensitive to the document fields; and as non-displaying fields ([0050] & [0065]), one or more path specifications corresponding to the document fields and to the value specification fields, said path specifications identifying nodes to be tested against completed value specifications. Second, receiving the selected document type and the completed value specifications and the corresponding path specifications. (*e.g.*, FIG. 21, ref 2104) Third, searching a subset of the self-describing, structured documents (2105) based on the completed value specifications and the corresponding path specifications, the subset including documents of the selected document type.

Claim 8 presents a computer-implemented method of searching a plurality of self-describing, structured documents, said documents including document fields. (OSpec, [0039], [0065]-0070)) The method includes four phases. First, providing a graphical user interface (*e.g.*, FIG. 5) including a document type selection filter (518); one or more document field selection filters (512), context sensitive to a selected document type; and one or more value specification fields (514), context sensitive to the document fields. Second, receiving the selected document type and the completed value specifications and document field identifiers corresponding to the completed value specifications. (*e.g.*, FIG. 21, ref 2104) Third, looking up path specifications corresponding to the document field identifiers, said paths specifications identifying nodes to be tested against completed value specifications. ([0065]) Fourth, searching a subset of the self-describing, structured documents based on the completed value specifications and the corresponding path specifications, the subset including documents of the selected document type. (2105, [0065])

Claim 24 presents a method of providing a searchable data base of self-describing, structured documents. (OSpec, [0044], [0046], [0066]-[0070]) This method includes three phases. First, loading a set of document type and path specification pairs (2404), said path specifications identifying nodes of documents to be indexed and searched. Second, indexing portions of the documents (*e.g.*, FIG. 26, [0070]) corresponding to the document type and path specification pairs. Third, providing a graphical user interface (*e.g.*, FIG. 5) including a document type selection filter (518); one or more document field selection filters (512), context sensitive to a selected document type; one or more value specification fields (514), context sensitive to the

document fields; and as non-displaying fields ([0050] & [0065]), one or more aliases to path specifications corresponding to the document fields and to the value specification fields, said paths specifications identifying nodes of the documents to be tested against completed value specifications.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The rejections under § 101 were withdrawn in the Final Office Action (FOA), as confirmed by the Examiner on p. 2 of the Advisory Action (AA).

Remaining for appeal is the issue of whether it was improper to reject independent claims 1, 8 and 24 under 35 U.S.C. 103(a) as being unpatentable over Probst, in view of Harold?

VII. ARGUMENT

In these rejections, Applicants fear that the Examiner considers the claimed technology to be so elegant that he should not have to search for a teaching or suggestion to combine the references to produce the claimed combination. The Examiner writes, in FOA at 15, "[T]he Office has addressed Applicant's arguments regarding Probst and Harold above. There's nothing novel about employing well known standards or languages in the programming arts. It's merely a matter of obvious design choice as to what programming languages and standards one chooses to employ within any software/system development effort." The Examiner's opinion, without any supporting examiner's declaration or reference to prior art, runs contrary to the "as a whole" rule of Section 103 and is sometimes called hindsight. The Federal Circuit explained in *Ruiz v. A.B. Chance*, 357 F.3d 1270, 1275, 69 U.S.P.Q.2d (BNA) 1686 (Fed. Cir. 2004):

In making the assessment of differences, section 103 specifically requires consideration of the claimed invention "as a whole." Inventions typically are new combinations of existing principles or features. *Envtl. Designs, Ltd. v. Union Oil Co.*, 713 F.2d 693, 698 (Fed. Cir. 1983) (noting that "virtually all [inventions] are combinations of old elements."). The "as a whole" instruction in title 35 prevents evaluation of the invention part by part. Without this important requirement, an obviousness assessment might break an invention into its component parts (A + B + C), then find a prior art reference containing A, another containing B, and another containing C, and on that basis alone declare the invention obvious. This form of hindsight reasoning, using the invention as a roadmap to find its

prior art components, would discount the value of combining various existing features or principles in a new way to achieve a new result - often the very definition of invention.

Section 103 precludes this hindsight discounting of the value of new combinations by requiring assessment of the invention as a whole. This court has provided further assurance of an "as a whole" assessment of the invention under § 103 by requiring a showing that an artisan of ordinary skill in the art at the time of invention, confronted by the same problems as the inventor and with no knowledge of the claimed invention, would select the various elements from the prior art and combine them in the claimed manner. In other words, the examiner or court must show some suggestion or motivation, before the invention itself, to make the new combination. *See In re Rouffet*, 149 F.3d 1350, 1355- 56 (Fed. Cir. 1998).

See, Princeton Biochemicals, Inc. v. Beckman Coulter, Inc., 411 F.3d 1332, 1337, 75 U.S.P.Q.2d (BNA) 1051 (Fed. Cir. 2005) (reciting *Ruiz* rule; "simply identifying all of the elements in a claim in the prior art does not render a claim obvious"). The Federal Circuit has rejected the Examiner's general approach, both as lacking the required evidentiary support (*In re Lee*, 277 F.3d 1338, 1342-44, 61 USPQ2d at 1433-34 (Fed. Cir. 2002); *In re Kotzab*, 217 F.3d 1365, 1369-70 (Fed. Cir. 2000); *Kolmes v. World Fibers Corp.*, 107 F.3d 1534, 1541 (Fed. Cir. 1997)) and because the logic applied is prohibited by statute (*Ruiz*; *Princeton Biochemicals*).

A. Preliminary Review of the Technology Disclosed and References

Before discussing the rejections that prompt this appeal, it is helpful to review the technology disclosed in the application and the references. With the divergent technologies in mind, this case should be easy to decide.

1. The Disclosed Technology

This application discloses what might be called light-weight XML document handling technologies. Light-weight technologies are useful to businesses that have not developed a full-fledged, customized system for handling XML document messages.

The disclosure of this application is common to this and two related applications. In both of the pending related applications, No. 10/026,364, METHOD AND APPARATUS FOR DECLARATIVE UPDATING OF SELF-DESCRIBING, STRUCTURED DOCUMENTS, by inventors Muljadi Sulistio, Yang Wei, Kelly Lane

Schwarzhoff and Yuan Din; and No. 10/026,366, METHOD AND APPARATUS FOR DECLARATIVE ERROR HANDLING AND PRESENTATION, by inventors Muljadi Sulistio, Yang Wei, Kelly Lane Schwarzhoff and Yuan Din, Applicants recently have received notices of allowance after an in-person interview with Examiner Kyle Stork.

Light-weight technologies are better adapted to document exchanges than editors and less expensive to develop than fully integrated, customized systems. They are useful, for instance, to small machine shops in the Detroit area that supply parts to automobile manufacturers. Small shops can meet the IT requirements of their customers by processing a few XML documents using light-weight technologies. They also are useful to large organizations when piloting a new document flow.

The light-weight technology of claim 1 includes a search interface that leads a user through building a search. Practically speaking, the user selects a document type, then selects among document fields that are included in that document type. The user enters a search string in a value specification field that is context sensitive to the document field selected. While the claim does not say this, one of skill in the art will appreciate that typical XML document schemas, such as SOX or WSDL, allow the interface to be based directly on the schema for the selected document type. (Similar graphical interfaces appear in claims 8 and 24.) The system that provided the user interface receives the user's selections with embedded path specifications. The system performs a search.

Claim 8 substitutes document field identifiers for path specifications and includes using the field identifiers instead of receiving embedded path specifications. The system performs a search.

Claim 24 addresses building a searchable database, then providing a search interface, without requiring the step of performing as search.

2. The Probst Reference

Probst describes a multi-media asset searching system. The graphical interface on which the Examiner relies appears as figure 5, set out on the next page and explained in Probst [0050]. The example that Probst gives for key words (502) is "Oscar belt", which would find a picture that has the metadata "Oscar Delohoya wearing a championship belt." The examples of possible asset types (503) given in [0050] are photos, audio, video and text. Only a single key word search field is provided by the

graphical interface, regardless of the asset type. The key word search field is not context sensitive to a document type or to a document field that relates to the document type – asset metadata is searched without distinction among fields, even though Probst's Table 1 identifies numerous types of metadata that could be maintained in separate fields.

3. The Harold Reference

Harold is an XML primer. The

Examiner selected glossary entries from chapter 1 (pp. 14-15) and other pages from chapters 2 (pp. 39-42) and 9 (pp. 259-271). We do not find any reference to the glossary entries in the Examiner's FOA. Primary reliance is placed on pp. 259-60, 264 and 268. Those pages explain XPointers, particularly an introduction to XPointers (259-260), relative location terms (264) and selection rules for selection by attribute (268). Overall, XPointer is a syntax for programmatically selecting part(s) of an XML document, as explained on p. 259:

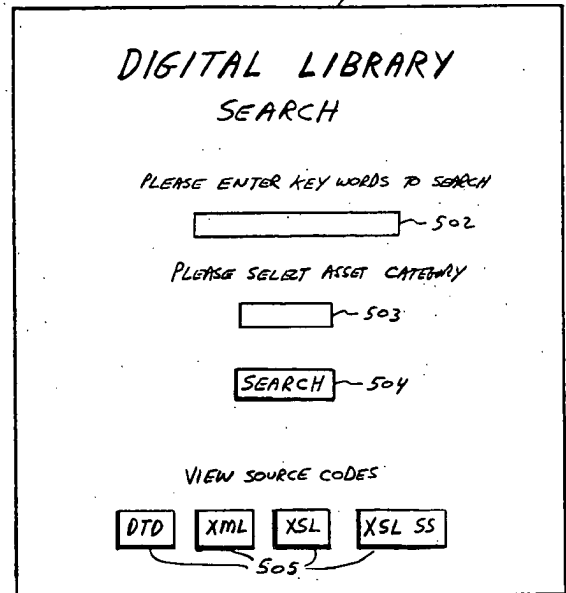
An XPointer can refer to a particular element of a document: to the first, the second, or the seventeenth such element, to the first element that's a child of a given element, and so forth. XPointers provide extremely powerful connections between documents without requiring the targeted document to contain additional markup just so its individual pieces can be linked to. [¶] Unlike HTML anchors, XPointers don't just point to a point in a document. They can point to ranges or spans. An XPointer might be used to select a particular part of a document ...

A sample XML document is given on pp. 261-63 and various XPointer language features are illustrated on 264-70 for selecting parts of the sample document.

None of the passages relied on from Harold apply XPointer to a graphic interface or to a context sensitive graphic interface.

4. The XPath Spec Reference

The XPath Specification is relied on only with respect to dependent claims, which we have grouped in this appeal, with independent claims.



With these divergent technologies in mind, the Board should find these claims easy to address.

B. Rejection of Independent Claim 1 Under Section 103(a) as Unpatentable over Probst in view of Harold was Improper

Claim 1 includes the limitations:

providing a graphical user interface including

a document type selection filter;

one or more document field selection filters, context sensitive to a selected document type;

one or more value specification fields, context sensitive to the document fields;

These limitations are not found in Probst. The Examiner relied on Probst figure references 503 and 705 and on paragraph [0056]. Looking first at figure 5, reproduced and explained above, entry of key words 502 (value specifications) is not context sensitive to an optional asset category 503 (document type). In a natural sequence, an asset category would optionally be entered after key words. There is no indication in the figure or the specification that any document field selection filters are provided or that value specification fields are context sensitive to selected document fields. To the contrary, Probst apparently searches all metadata fields for key words entered in 502, regardless of asset type. No provision is made for searching a particular metadata field (Table 1), much less for providing document field selection filter that is context-sensitive to document (asset) type.

Turning to figure 7, reference 705 does not represent part of a graphical user interface, as this figure depicts a DTD schema defining the structure of an XML document; it does not depict a graphical user interface. In paragraph [0056], Probst gives some preferred guidelines for an asset record schema, which are not related to any graphical user interface.

In the FOA at 14, the Examiner argued that "the asset category of FIG. 5 provides a context for key word searching." This argument does not meet the words of the claim. The claim calls for the user interface to provide document field selection filters, context sensitive to a selected document type and value specification fields, context sensitive to the document fields. The Examiner appears to be reading into Probst something that is not there. Probst figure 6 depicts results of a search across

asset types, returning 4 photos, 1 video and 1 audio, which is contrary to selecting a document/asset type first. In paragraph [0050], it appears that search fields 502 (key words) and 503 (asset category) are combined in a simple Boolean query (502 & 503) without either search field setting a context for the other. At best, the "context" for key word searching, to which the Examiner refers (FOA 14), resides in backend search logic and therefore does not meet the words of the claim, which refer to the front end user interface.

For these reasons, rejection of claim 1 should be reversed.

Claim 1 further includes the limitations:

as non-displaying fields, one or more path specifications corresponding to the document fields and to the value specification fields, said path specifications identifying nodes to be tested against completed value specifications

The Examiner relies on Harold to meet these limitations. But Harold at p. 268 and in chapter 9, generally, does not describe use of XPointers with any graphical interface. Harold's chapter 9 is more a language description than a teaching of how to apply the language. The concepts recapped on p. 271 are language concepts, not GUI applications of the language. In particular, Harold says nothing about using path specifications in non-displaying fields as a way of passing information from a GUI to a search engine.

Therefore, rejection of claim 1 based on Probst et al. in view of Harold should be reversed.

C. Rejection of Independent Claim 8 Under Section 103(a) as Unpatentable over Probst in view of Harold was Improper

The Examiner's rejection of **Claim 8** mirrors the rejection of claim 1, without hidden field limitations and with the following added limitations:

looking up path specifications corresponding to the document field identifiers, said paths specifications identifying nodes to be tested against completed value specifications;

searching a subset of the self-describing, structured documents based on the completed value specifications and the corresponding path specifications, the subset including documents of the selected document type.

The providing and receiving limitations are not found in Probst et al. in view of Harold,

for the reasons presented above, in the context of claim 1. Rejection of claim 8 should be reversed because the providing and receiving limitations, shared with claim 1, are not met by the proposed combination.

Probst et al. [0034] does not meet the searching limitations particular to claim 8 because the description of Advanced Search Screen (figure 1, ref 107) does not include the same detailed limitations as the claim. The Examiner relies on [0034] in FOA at 5. This passage refers to the Advanced Search Screen 107, which is actually described in [0029]:

[0029] At Advanced Search Screen 107 users enter or are given pre-defined criteria with which the database of the invention can be searched. Screen 107 preferably allows users to generate advanced search criteria including, but not limited to, searching specific strings, numeric values, fields or combinations of fields containing data, as well as allowing users to define complex or simple search expressions using boolean or other logic, wildcards, multiple search criteria, or any other type of search functionality. Preferably, upon an indication by a user or upon a pre-defined system event (for example, expiration of a timer); a user will be directed along path 119 to Search Results Screen 120.

As described, the advanced search capability does not include the claimed elements of *"searching a subset of the self-describing, structured documents based on the completed value specifications and the corresponding path specifications, the subset including documents of the selected document type"*. For this reason, rejection of claim 8 should be reversed.

The Examiner further relies on Harold at pp. 260, 264 and 268 to meet the looking up path specifications corresponding to the document field identifiers limitation. Harold does not meet this limitation because Harold is a general description of XPointers programming and does not apply that programming language to any GUI interface.

To argue for rejection, the Examiner extrapolates beyond anything found in Probst or Harold, using the present claims as a roadmap for applying prior art tools. Applicants see no teaching in either Probst or Harold to produce the claimed invention as a whole. *Ruiz v. A.B. Chance, supra*, 357 F.3d at 1275, 69 U.S.P.Q.2d (BNA) 1686 (section 103 specifically requires consideration of the claimed invention "as a whole."); *Princeton Biochemicals, Inc. v. Beckman Coulter, Inc., supra*, 411 F.3d at 1337, 75 U.S.P.Q.2d (BNA) 1051 (reciting *Ruiz* rule; "simply identifying all of the elements in a

claim in the prior art does not render a claim obvious"); *see also*, 2-5 Chisum on Patents § 5.03 [2][c] n. 29 (2005 Lexis version); *e.g. ATD Corp. v. Lydall, Inc.*, 159 F.3d 534, 546, 48 USPQ2d 1321, 1329 (Fed. Cir. 1998) ("Determination of obviousness can not be based on the hindsight combination of components selectively culled from the prior art to fit the parameters of the patented invention."); *Grain Processing Corp. v. American Maize-Products Corp.*, 840 F.2d 902, 907, 5 USPQ2d 1788, 1792 (Fed. Cir. 1988) ("Care must be taken to avoid hindsight reconstruction by using 'the patent in suit as a guide through the maze of prior art references, combining the right references in the right way so as to achieve the result of the claims in suit.'"). Accordingly, Applicants submit that the Examiner is relying on impermissible hindsight, and limitations not found in the combination relied upon, in rejecting the claims.

For these reasons, rejection of claim 8 based on Probst et al. in view of Harold should be reversed.

D. Rejection of Independent Claim 24 Under Section 103(a) as Unpatentable over Probst in view of Harold was Improper

Claim 24 includes the following additional limitations related to building a searchable database:

loading a set of document field and path specification pairs, said path specifications identifying nodes of self-describing, structured documents to be indexed and searched;

indexing portions of the documents corresponding to the document field and path specification pairs;

Other GUI-related limitations are not found in Probst in view of Harold, for the reasons given above, in the context of claim 1. Because the GUI-related limitations are not met, rejection of claim 24 should be reversed.

As a further part of his basis for rejecting claim 24, the Examiner relied on Probst figures 5 & 7 and [0016] and [0056]. The loading and indexing elements are not met by Probst figure 5, reference 503, which is part of a search GUI and not loading field and path specification pairs in preparation to index and search. These elements are not met by [0016] either, which reads in its entirety:

[0016] In accordance with this invention, data definitions are provided for digital assets that include a hierarchical structure that reflects the relationships between attributes and categories of content. These definitions, preferably encoded in XML, can be used as a standardized dictionary to create a digital asset library that is easily and economically

manageable. The data definitions are applicable to digital assets of disparate data types and include metadata identifiers sufficient to uniquely identify those digital assets.

Figure 7, reference 705, and [0056], which explains part of figure 7, do not meet these limitations because the figure depicts a schema and 705 is a data element within the schema. The content data element 705 may be considered a document field, but not a pair of document field and path specifications. Recall that Probst searches across all metadata in a single search, contrary to loading and indexing document fields and path specifications in the manner claimed. Because Probst is contrary, it cannot implicitly teach the claimed details of loading and indexing.

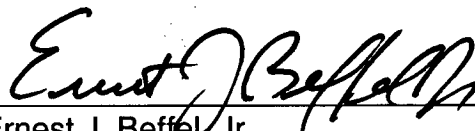
Therefore, rejection of claim 24 based on Probst in view of Harold should be reversed.

CONCLUSION

In view of the foregoing, Applicants/Appellants ask that this honorable Board reverse the Examiner's rejections of the claims. In addition, it is submitted that all claims are now allowable, and a notice of intent to issue a patent is respectfully requested.

The Commissioner is hereby authorized to charge any fee determined to be due in connection with this communication, or credit any overpayment, to our Deposit Account No. 50-0869 (Attorney Docket No. JGR 1009-1).

Respectfully submitted,


Ernest J. Boffel, Jr.
Registration No. 43,489
Attorney for Patent Owner

Dated: 29 December 2005

HAYNES BEFFEL & WOLFELD LLP

P.O. Box 366
751 Kelly Street
Half Moon Bay, CA 94019
Telephone: 650.712.0340
Facsimile: 650.712.0263

CLAIMS APPENDIX

1. (Previously presented) A computer-implemented method of searching a plurality of self-describing, structured documents, said documents including document fields, the method including:
 - providing a graphical user interface including
 - a document type selection filter;
 - one or more document field selection filters, context sensitive to a selected document type;
 - one or more value specification fields, context sensitive to the document fields; and
 - as non-displaying fields, one or more path specifications corresponding to the document fields and to the value specification fields, said path specifications identifying nodes to be tested against completed value specifications;
 - receiving the selected document type and the completed value specifications and the corresponding path specifications; and
 - searching a subset of the self-describing, structured documents based on the completed value specifications and the corresponding path specifications, the subset including documents of the selected document type.
2. (Original) The method of claim 1, wherein the path specifications are compliant with any version of an XPath standard.
3. (Original) The method of claim 1, wherein the self-describing, structured documents are compliant with any version of an XML standard.
4. (Original) The method of claim 3, wherein the self-describing, structured documents are compliant with any version of an XML standard.
5. (Previously presented) The method of claim 1, wherein the graphical user interface is a character string compliant with any version of an HTML standard.

6. (Previously presented) The method of claim 3, wherein the graphical user interface is a character string compliant with any version of an HTML standard.
7. (Previously presented) The method of claim 4, wherein the graphical user interface is a character string compliant with any version of an HTML standard.
8. (Previously presented) A computer-implemented method of searching a plurality of self-describing, structured documents, said documents including document fields, the method including:
 - providing a graphical user interface including
 - a document type selection filter;
 - one or more document field selection filters, context sensitive to a selected document type; and
 - one or more value specification fields, context sensitive to the document fields;
 - receiving the selected document type and the completed value specifications and document field identifiers corresponding to the completed value specifications;
 - looking up path specifications corresponding to the document field identifiers, said paths specifications identifying nodes to be tested against completed value specifications; and
 - searching a subset of the self-describing, structured documents based on the completed value specifications and the corresponding path specifications, the subset including documents of the selected document type.
9. (Original) The method of claim 8, wherein the path specifications are compliant with any version of an XPath standard.
10. (Original) The method of claim 8, wherein the self-describing, structured documents are compliant with any version of an XML standard.

11. (Original) The method of claim 10, wherein the self-describing, structured documents are compliant with any version of an XML standard.
12. (Previously presented) The method of claim 8, wherein the graphical user interface is a character string compliant with any version of an HTML standard.
13. (Previously presented) The method of claim 10, wherein the graphical user interface is a character string compliant with any version of an HTML standard.
14. (Previously presented) The method of claim 11, wherein the graphical user interface is a character string compliant with any version of an HTML standard.
15. (Previously presented) A method of specifying where to search among a plurality of self-describing, structured documents, said documents having document types and including document fields, the method including:
 - displaying a graphical user interface including
 - a document type selection filter;
 - one or more document field selection filters, context sensitive to a selected document type; and
 - one or more value specification fields, context sensitive to the document fields;
 - the graphical user interface further including, as non-displaying fields, one or more path specifications corresponding to the document fields and to the value specification fields, said paths specifications identifying nodes in the documents to be tested against completed value specifications;
 - receiving from a user the selected document type and the completed value specifications; and
 - transmitting to a server the selected document type and the completed value specifications and the path specifications corresponding to the completed value specifications.

16. (Previously presented) A computer-implemented graphical user interface, including:
- a document type selection filter;
 - one or more document field selection filters, context sensitive to a selected document type;
 - one or more value specification fields, context sensitive to the document fields; and
 - as non-displaying fields, one or more path specifications corresponding to the document fields and to the value specification fields, said paths specifications identifying nodes of a self-describing, structured document to be tested against completed value specifications.
17. (Original) The method of claim 16, wherein the path specifications are compliant with any version of an XPath standard.
18. (Original) The method of claim 16, wherein the self-describing, structured documents are compliant with any version of an XML standard.
19. (Original) The method of claim 18, wherein the self-describing, structured documents are compliant with any version of an XML standard.
20. (Previously presented) The method of claim 16, wherein the graphical user interface is a character string compliant with any version of an HTML standard.
21. (Previously presented) The method of claim 18, wherein the graphical user interface is a character string compliant with any version of an HTML standard.
22. (Previously presented) The method of claim 19, wherein the graphical user interface is a character string compliant with any version of an HTML standard.
23. (Previously presented) A method of providing a searchable data base of self-describing, structured documents, including:

loading a set of document field and path specification pairs, said path specifications identifying nodes of self-describing, structured documents to be indexed and searched;

indexing portions of the documents corresponding to the document field and path specification pairs; and

providing a graphical user interface based on the set, including

a document type selection filter;

one or more document field selection filters, context sensitive to a selected document type;

one or more value specification fields, context sensitive to the document fields; and

as non-displaying fields, one or more path specifications corresponding to the document fields and to the value specification fields, said paths specifications identifying nodes of the documents to be tested against completed value specifications.

24. (Previously presented) A method of providing a searchable data base of self-describing, structured documents, including:

loading a set of document type and path specification pairs, said path specifications identifying nodes of documents to be indexed and searched;

indexing portions of the documents corresponding to the document type and path specification pairs; and

providing a graphical user interface including

a document type selection filter;

one or more document field selection filters, context sensitive to a selected document type;

one or more value specification fields, context sensitive to the document fields; and

as non-displaying fields, one or more aliases to path specifications corresponding to the document fields and to the value specification fields, said paths specifications identifying nodes of the documents to be tested against completed value specifications.